Page 4

Visual ordnance recognition for clearing test ranges

Clark F. Olson and Larry H. Matthies

Jet Propulsion Laboratory, California Institute of Technology

4800 Oak Grove Drive, Mail Stop 107-102, Pasadena, CA 91109

ABSTRACT

We describe a method for recognizing surface-lying ordnance in test ranges using stereo range information and image edge maps. This method is to be used by an unmanned ground vehicle (UGV) surveying the test range for autonomous clearing of ordnance. We concentrate on a particular type of cylindrical ordnance (BLU-97) in current usage in U.S. military test ranges. In order to locate instances of the ordnance, we employ a stereo pair of cameras to be mounted on top of a UGV. Parallel segments corresponding to the occluding contours of the ordnance are detected in the imagery using robust and efficient model extraction techniques. The stereo range data is used to adaptively select the local scale for edge detection and to place constraints on the search space for the parallel segment extraction. Initial tests indicate that robust recognition is possible in near real-time with a low rate of false positives.

Keywords: Ordnance recognition, stereo vision, edge detection, adaptive scale selection, parallel line extraction, cylinder detection, autonomous munitions clearance

1. INTRODUCTION

Military test ranges containing unexploded ordnance due to live-fire testing and training exercises have become a significant safety problem in many locations. It has been estimated that over 800 locations and 11 million acres in the United States alone have been potentially contaminated with unexploded ordnance.[1] The cleanup of these sites has become of particular importance in this era of base realignment and closure. The estimated cleanup of sites in this category alone is over $4 billion. Furthermore, cleanup of such sites by human technicians is dangerous and the loss of human lives adds immeasurably to the cost.

We are developing an ordnance recognition method for use on an unmanned ground vehicle (UGV) in conjunction with the Rough Terrain Surface Munitions Clearance project at Wright Laboratory, Tyndall Air Force Base. The envisioned system consists of an unmanned all-terrain vehicle with on-board stereo cameras, in addition to verification and neutralization devices. The vehicle will either survey the test range, using the cameras as a pushbroom-like detection device, or move to a designated observation point and sweep the area of interest by panning and tilting the cameras. This paper describes a method for recognizing surface-lying ordnance using such a system.

Further author information: http://robotics.jpl.nasa.gov/people/olson/homepage.html

Figure 1. Image of BLU-97 ordnance acquired at a Nellis Air Force Base test range.

We concentrate on the recognition of BLU-97 ordnance, which is in current usage in U.S. military test ranges (Figure 1). The body of this type of ordnance is cylindrical in shape, approximately 20 centimeters long, and has a 6 centimeter diameter. Our strategy is thus to seek cylindrical objects of the appropriate size in visual imagery. In order for the system to be effective in this application, we estimate that the system should be able to recognize ordnance at least 10 meters from the vehicle and that a 5 meter swath width at this range is desirable (implying a lens with approximately 30° field-of-view), as well as a latency below 5 seconds. The 5 meter swath width implies that a frontal view of the ordnance at the maximum range in a 500×500 image yields a 20 pixel long strip in the image and non-frontal views will be smaller.

The method that we use to detect the ordnance is based on finding pairs of parallel segments in the image edge map. These segments correspond to the occluding contours of the cylindrical body of the ordnance. We first smooth the image and perform edge detection adaptively, using stereo range information[2] to select the appropriate scale for smoothing and the appropriate thresholds for edge detection at each pixel.[3] Pairs of parallel segments are then detected using efficient and robust model extraction techniques.[4] These techniques sample a series of oriented edge pixels from the edge map of image. For each sampled edge pixel, pose space clustering techniques are used to determine if the pixel belongs to some pair of parallel segments in a local neighborhood in the image edge map. Robust detection can be achieved after a relatively small number of edge pixels have been sampled.

Constraints on the distance to each sampled pixel in the image are derived using the stereo range information and these constraints are used in the detection process to limit the search space that is required in each trial to detect the ordnance, if it is present. These constraints reduce both the likelihood of finding false positive instances and the amount of computation required by the search process.

This method has been implemented on a workstation and empirical experiments indicate that we can achieve a high rate of success in finding the ordnance with a low false alarm rate through the use of these techniques. It is expected that a near-real time implementation of these techniques is realizable with off-the-shelf computing hardware.

2. STEREO RANGE DATA

The use of stereo range data is crucial to the ordnance recognition system that we have developed. We use this information for two purposes:

1. It allows us to select the appropriate scale at each location in the image for smoothing and edge detection.
2. It allows us to place bounds on the size of the ordnance at any location in the image, thus reducing both the search space and the number of false positives found.

The techniques that we use to compute the stereo range data have been described elsewhere.[2,5] The important points are summarized here.

An off-line step, where the stereo camera rig is calibrated must first be performed. We use a camera model that allows arbitrary affine transformation of the image plane[6] and that has been extended to include radial lens distortion.[7] The remainder of the method is performed on-line.

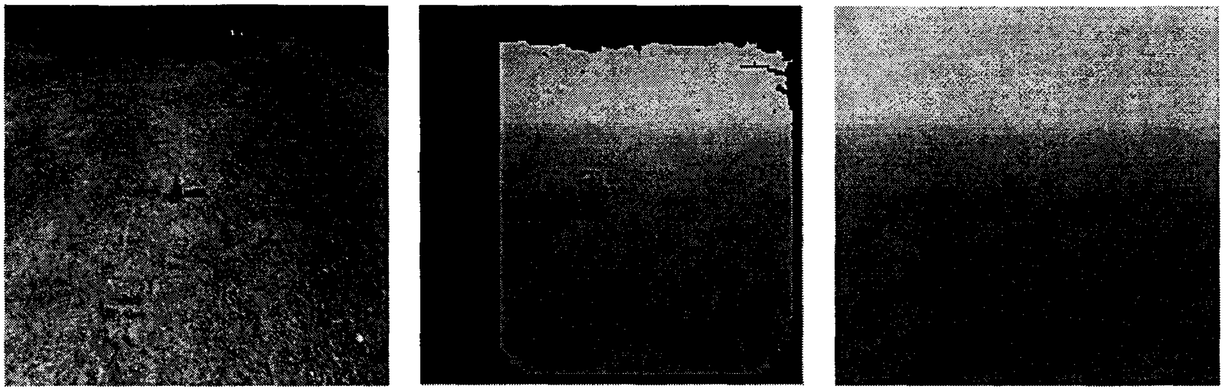
At run-time, each image is first warped to remove the lens distortion and the images are rectified so that the corresponding scan-lines yield corresponding epipolar lines in the image. The disparity between the left and right images is measured for each pixel by minimizing the sum-of-squared-difference (SSD) measure of a window around the pixel in the Laplacian of the image. Subpixel disparity estimates are computed using parabolic interpolation on the SSD values neighboring the minimum. Outliers are removed through consistency checking and smoothing is performed over a 3×3 window to reduce noise. Finally, the coordinates of each pixel are computed using triangulation.

Note that not every pixel is assigned a range with this method. There are a variety of factors that result in some pixels not being assigned a range including occlusion, window effects, finite disparity limits, and outliers. Despite this problem, we desire a range estimate at each pixel in the image that is designated an edge in order to perform ordnance detection efficiently. To resolve this problem, we propagate the range values from neighboring pixels using a simple method that prefers neighbors to the left or right to those above or below.

Figure 2 shows an example of the range data computed using these techniques. A real-time implementation of these techniques runs at approximately two frames per second.[5]

3. VARIABLE-SCALE EDGE DETECTION GUIDED BY STEREOSCOPY

In order to detect the ordnance, we search for parallel segments in the image edge map of the left image of the stereo pair. It is thus crucial to extract the edges corresponding to the ordnance in a robust manner. It is also desirable to keep the frequency of clutter



(a) (b) (c)

Figure 2. Range data extracted from a stereo pair. (a) Left image of a stereo pair. (b) Distance from the camera mapped into gray values. Black pixels indicate no valid range data. (c) Range map after filling unknown values with estimates.

edges as low as possible, so that there is less data that needs to be processed during detection and so that the probability of a false positive detection is small. Unfortunately, the terrain of interest often yields cluttered edge maps due to the presence of rocks, textured soil, and other phenomena. To alleviate this problem, we perform smoothing adaptively, such that the scale of the filter varies with the range from the camera and is tuned to the approximate size of the ordnance at that range.

Our approach to performing adaptive smoothing is to filter the image at multiple scales and then interpolate the response for each pixel at the appropriate scale given by the range data. Since we adapt the Canny edge detector[8] for this application, we use Gaussian derivative filters to perform the smoothing (and differentiation) of the image. If the response at each pixel is appropriately normalized, the resulting image can be processed as in the typical edge detector. For the Canny edge detector, this is performed through non-maxima suppression and a hysteresis thresholding procedure.

We must first select an appropriate set of scales to use. Based on the size of the ordnance and the ranges over which we desire accurate recognition in the test imagery, we have chosen to work with scales in the range $0.8 \leq \sigma \leq 3.2$, and we perform smoothing at three scales ($\sigma_1 = 0.8, \sigma_2 = 1.6, \sigma_3 = 3.2$) in order to interpolate any scale in the range accurately. At each scale, we filter with the Gaussian derivative in both x and y. Each image is thus convolved with 6 filters.

Now, we must interpolate the gradient at each pixel for the appropriate scale given by the stereo range information. Since there is an inverse linear relationship between the range to an object and its scale in the image, we map the range information into a scale for smoothing and edge detection using:

$$\sigma(x, y) = K/R(x, y), \tag{1}$$

where $R(x, y)$ is the range estimate given by the stereo processing at the pixel, and K is a constant determined by the size of the object of interest. For this case, we use $K = 4$.

We approximate the correct response at each pixel using parabolic interpolation (separately for x and y) in the $\ln \sigma$ domain. Let $F_\sigma(x, y)$ be the desired response at (x, y) for the scale σ. In determining the equation that yields the appropriate response, it is useful to perform a coordinate transform such that $z = \log_2 \frac{\sigma(x,y)}{\sigma_2}$. For $\sigma_1 = \frac{1}{2}\sigma_2 = \frac{1}{4}\sigma_3$, this yields $z_1 = -1$, $z_2 = 0$, and $z_3 = 1$. With this transformation it is simple to show that:

$$
\begin{aligned}
F_\sigma(x, y) &\approx az^2 + bz + c && (2)\\
a &= \frac{1}{2}(F_{\sigma_1}(x, y) - 2F_{\sigma_2}(x, y) + F_{\sigma_3}(x, y)) && (3)\\
b &= \frac{1}{2}(F_{\sigma_3}(x, y) - F_{\sigma_1}(x, y)) && (4)\\
c &= F_{\sigma_2}(x, y) && (5)\\
z &= \log_2 \frac{\sigma(x, y)}{\sigma_k} && (6)
\end{aligned}
$$

Following the variable-scale smoothing described above, we proceed with Canny's edge detection method[8] on the smoothed image. This technique computes the image gradients over the image in the x- and y-directions in order to determine the orientation and magnitude of the gradient at each pixel. Note, however, that if the gradient magnitudes are to be comparable, we must normalize them. This can be easily be seen by noticing that the response of a step edge to a Gaussian derivative filter varies with the scale of the filter. A 1-dimensional Gaussian derivative perfectly aligned with a step edge yields a response proportional to $\frac{1}{\sigma}$. To correct this problem, we normalize the gradient magnitude at each pixel by multiplying by $\sigma(x, y)$.

Finally, non-maxima suppression is performed and the edges are thresholded using hysteresis thresholding. We determine the hysteresis thresholds adaptively through examination of the histogram of gradient magnitudes.

Figure 3 shows an example of the edge detection techniques and a comparison with a non-adaptive version of the edge detector. For this image, the variable scale smoothing was performed over the range $0.8 \leq \sigma \leq 3.2$, while the non-adaptive version used a scale of $\sigma = 1.6$. Note that, when variable-scale smoothing was used, more of the ordnance edges were detected and with greater accuracy, while less texture edges are found on the ground.

In addition to its use in performing edge detection, stereoscopy is helpful in determining edge salience. Shorter edges that are detected at a larger distance are more likely to correspond to salient world edges than edges at close range that appear to be long due to perspective effects. We use the summed gradient magnitude over the length of the edge and the local straightness of the edge as salience criteria,[3] although many other salience measures could be used.[9]

(a) (b) (c)

Figure 3. Edges are detected using variable scale smoothing and detection. (a) Example grey-scale image. (b) Edges detected with variable scale smoothing. (c) Edges detected with typical Canny edge detector.

4. DETECTING PARALLEL SEGMENTS

Once the edges have been detected, we seek the pairs of parallel edge segments that correspond to the occluding contours of the cylindrical ordnance body. The algorithm that we use to detect these segments is an instance of the RUDR paradigm.[4] RUDR is a general method to perform model extraction and fitting that has been previously used to perform curve detection,[10] object recognition,[11] and robust regression.[12] The main components of the paradigm are the decomposition of the problem into many smaller subproblems, the use of randomization to limit the number of subproblems that must be examined, and the use of pose space analysis techniques to solve each subproblem.

The first step in applying the RUDR paradigm to a particular problem is to determine the model and the type of data features that will be used. For the detection of parallel lines, our model is simply a parameterization of the parallel lines. We use the (ρ, θ) parameterization for the lower line:

$$\rho = x \cos\theta + y \sin\theta, \tag{7}$$

along with a parameter, d, describing the perpendicular distance to the upper line. The edge pixels detected as described above (along with the image gradients at the same locations) are used as our data features.

In the noiseless case, two oriented edge pixels overconstrain the position of a pair of parallel lines (assuming one pixel lies on each line). When noise is considered, a subset of these pairs of pixels will be consistent with belonging to parallel segments. A pose space technique (such as variants of the Hough transform) could detect the parallel segments by computing the parallel line positions given by each pair of edge pixels (eliminating those pairs that not consistent with any pair of parallel lines) in the image and seeking parameters in the space of parallel lines, (ρ, θ, d), that are consistent with many of the pairs of pixels. The RUDR paradigm uses similar ideas, except that the problem is divided into many smaller subproblems by considering (in this case) a sequence of *distinguished pixels* from among the edge pixels. Each distinguished pixel is simply a randomly chosen edge pixel that has been chosen to constrain the current subproblem. Each subproblem is constrained to examine only those pairs of pixels that include the distinguished pixel. It has been shown that, with precise pose space analysis techniques, this decomposition of the problem yields the same results as standard Hough transform techniques if each possible distinguished pixel is examined.[4]

Analysis has shown that an arbitrarily low probability of failure γ can be achieved through the examination of approximately $\frac{-\ln \gamma}{f}$ distinguished pixels, where f is the minimum fraction of the data features that must belong to the parallel lines to detect them robustly.[4] Each subproblem is examined in $O(n)$ time, where n is the number of data features, as discussed below.

Now, for each distinguished pixel that is examined, we must determine if it belongs to some pair of parallel lines in the image edge data. We assume that the localization error for each edge pixel can be bounded by some small error and we have empirically chosen a maximum allowable error in the position of each edge pixel to be $\frac{1}{2}$ pixel and the maximum allowable error in the orientation of each edge pixel to be $\frac{\pi}{20}$.

Our strategy is to project the space of parameter positions (if any) that are consistent (up to the bounded error) with each pair of oriented pixels that include the distinguished pixel onto the manifold in the parameter space that is consistent with the distinguished pixel in the errorless case. If many of these projections overlap at some point on this manifold, this indicates that a segment exists opposite the distinguished pixel with the correct orientation. If the distinguished pixel also lies on a segment with the appropriate orientation, then a pair of parallel segments has been found.

Since the distinguished pixel completely specifies the location of one of the lines in the errorless case, the remaining degree of freedom is the distance of second line. The manifold we are projecting onto is thus a curve in the three-dimensional parameter space that can be parameterized by the distance from the line given by the (oriented) distinguished pixel. In this case, the projection is simply the range of the possible distances between the two parallel lines that are consistent with the pair of pixels. Note that this distance may be negative, since we do not know which of the lines, if either, the distinguished pixel lies on.

In order to determine the projection discussed above, we consider the set of orientations that are consistent with both of the pixels up to the given error boundaries. Each orientation in this set yields a distance between the pair of parallel lines. If the two pixels under consideration have coordinates (x_i, y_i) and (x_j, y_j), then a pair of parallel lines passing through them with orientation θ are given by:

$$y - y_i = (x - x_i) \tan \theta \tag{8}$$

and

$$y - y_j = (x - x_j) \tan \theta. \tag{9}$$

The perpendicular distance between the lines is:

$$d = \frac{(y_i - y_j) - (x_i - x_j)\tan\theta}{\sqrt{1 + \tan^2\theta}}. \tag{10}$$

The feasible range of such distances can be determined by considering the minimum and maximum orientation consistent with the two pixels and, if the set of consistent orientations for the parallel lines contains the orientation of the line perpendicular to the segment between the pixels, the (signed) distance between the pixels. To this range we must also add the allowable localization error in the positions of the pixels. This final range is our projection of the pose space that is consistent with the pair of pixels onto the curve in the parameter space consistent with the distinguished pixel.

The distances at which many of these projections intersect yield the positions of the opposing segments that are parallel to the orientation of the distinguished pixel. We detect these segments by discretizing the range of possible distances into 1 pixel wide intervals and maintaining a counter for each interval. For each of the distance ranges that is given by a pair of pixels, the counter is incremented for each interval that the range overlaps. After the distinguished pixel has been paired with each additional pixel in some neighborhood, we look for peaks in the counter array.

In addition to seeking the segments that oppose the distinguished pixel, we use the error bounds to determine how many of the pixels lie on the same segment as the distinguished pixel. If both pixels have orientations that are consistent with being perpendicular to the segment between the pixels, then the pixels are considered to be parallel up to the localization error and this count is incremented.

Finally, we must have some criterion determining which parallel segments are output. We report all of the parallel segments for which both the count on the number of pixels on a segment with the distinguished pixel and the count on the number of pixels on some segment parallel to this segment surpasses some predetermined fraction of the estimated ordnance length in the image at this position (see below).

4.1. Placing Bounds on the Ordnance Size

In order to improve the efficiency of the method and reduce the rate of false positives, we use the stereo range information to place bounds on the size of the ordnance given the hypothesis that the distinguished pixel lies on the occluding contour of the ordnance body. The length and width estimate for the ordnance (assuming a frontal view) are given by:

$$L_{\text{image}} = \frac{L_{\text{true}}}{R\sqrt{\left(\frac{\cos\theta}{H}\right)^2 + \left(\frac{\sin\theta}{V}\right)^2}} \tag{11}$$

and

$$W_{\text{image}} = \frac{W_{\text{true}}}{R\sqrt{\left(\frac{\sin\theta}{H}\right)^2 + \left(\frac{\cos\theta}{V}\right)^2}}, \tag{12}$$

where R is the range estimate of the ordnance (given by the stereo range information at the distinguished pixel), θ is the orientation along the ordnance axis in the image, L_{true} and W_{true} are the true length and width of the ordnance, and H and V are horizontal and vertical scaling factors in the image.*

When the ordnance is oriented such that the axis is not horizontal in the image, the length of the ordnance in the image will be less than L_{image} due to foreshortening. The foreshortened length is estimated under the assumption that the ordnance lies flat on a horizontal ground plane as follows:

$$L_{\text{foreshortened}} = L_{\text{image}} \left(\frac{H}{R} + \left(1 - \frac{H}{R}\right) |\cos\theta|\right), \tag{13}$$

where H is the height of the cameras above the ground plane.

This information is used to determine what threshold should be used in detecting the parallel lines. In addition, we use this information to improve the performance of the method. For each distinguished pixel, we need only consider the other pixels that are within d_{ordnance} of the distinguished pixel, where d_{ordnance} is given by:

$$d_{\text{ordnance}} = \sqrt{L^2_{\text{foreshortened}} + W^2_{\text{image}}}. \tag{14}$$

In addition, only those hypotheses with width close to W_{image} are valid hypotheses, the rest need not be considered.

5. RESULTS

This method has been applied to a set of 48 pairs of stereo, greyscale images. These images were captured at two locations near the Jet Propulsion Laboratory, using two camera orientations at each location for a total of four settings. In each setting, the cross-product of 4 ordnance distances (1, 2, 3, and 4 meters) and 3 ordnance orientations (0°, 45°, and 90°) with respect to the camera orientation were imaged. Relatively short distances were used since an 85° field-of-view lens was used for this dataset. In practice, we expect to use a field-of-view closer to 30°, so the distance should scale approximately as a factor of 3.

Overall, 40 of the 48 ordnance instances were detected with a total of 18 false positives. All of the false negative detections occurred for ordnance at the 90° orientation and at a distance of 3 or 4 meters (all of these cases failed, while none of the others failed). The 90° orientation is unfavorable for ordnance recognition since foreshortening of the ordnance results in fewer edge pixels being detected along the occluding boundary. We conclude that these techniques are capable of detecting the ordnance as long as either

*We do not assume aspect ratio of 1:1.

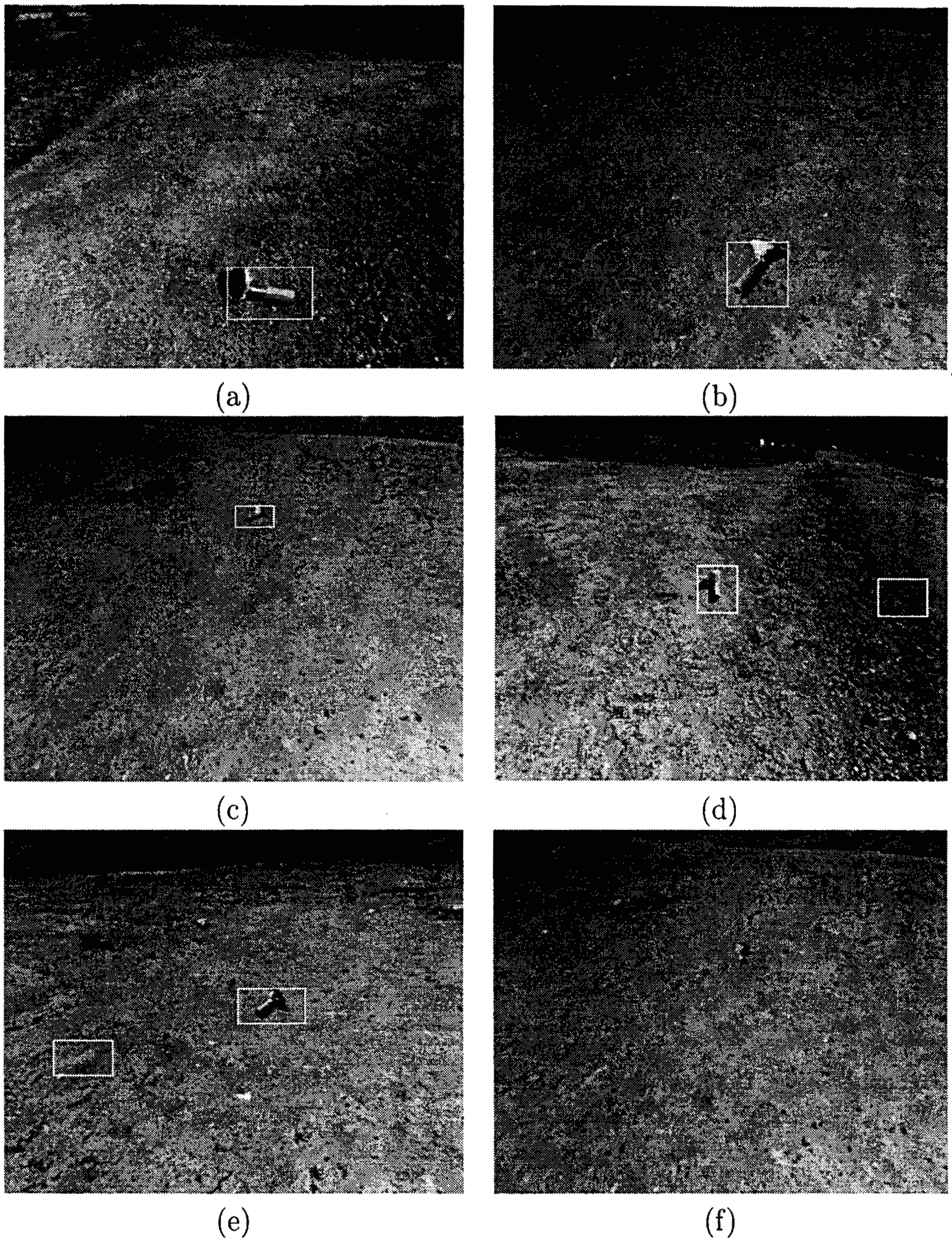


Figure 4. Examples of recognition results. (a) Correct detection at close range. (b) Another correct detection at close range. (c) Correct detection at long range. (d) Correct detection with a false positive. (e) Correct detection with a false positive. (f) False negative due to unfavorable orientation and distance.

page 5

(1) the ordnance is not oriented unfavorably (nearly along the camera axis) or (2) the ordnance is not far from the camera. Figure 4 shows several examples, including two false positives and a false negative.

6. SUMMARY

We have described a method to detect surface-lying BLU-97 ordnance in a test range using computer vision. The method uses range data from a stereo pair of cameras and image edge data to detect the cylindrical ordnance in the images. The stereo range data enhances the edge detection process, allowing the scale of smoothing and edge detection to be set adaptively according to the range, and reduces the search space by placing constraints on the size of the ordnance in the image. Pairs of parallel segments are extracted from the edge data corresponding to the occluding boundaries of the cylinders. Good results have been demonstrated on an initial test set of greyscale imagery. Our future plans are to extend these techniques to color imagery, where we can also use the ordnance color as a recognition cue and more extensive testing on a data set collected at a live-fire test range.

7. ACKNOWLEDGMENTS

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored by the Wright Laboratory at Tyndall Air Force Base, Panama City, Florida, through an agreement with the National Aeronautics and Space Administration.

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government, or the Jet Propulsion Laboratory, California Institute of Technology.

REFERENCES

1. C. M. Mackenzie, C. Jordan, R. E. Dugan, and M. A. Kolodny, "Detecting UXO - Putting it all into perspective," in *Detection Technologies for Mines and Minelike Targets, Proc. SPIE 2496*, pp. 94–99, 1995.
2. L. Matthies, "Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation," *International Journal of Computer Vision* **8**, pp. 71–91, July 1992.
3. C. F. Olson, "Variable-scale smoothing and edge detection guided by stereoscopy," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1998.
4. C. F. Olson, "RUDR: A general paradigm for model extraction and fitting." In progress.
5. L. Matthies, A. Kelly, T. Litwin, and G. Tharp, "Obstacle detection for unmanned ground vehicles: A progress report," in *Proceedings of the International Symposium on Robotics Research*, pp. 475–486, 1996.
6. Y. Yakimovsky and R. Cunningham, "A system for extracting three-dimensional measurements from a stereo pair of TV cameras," *Computer Vision, Graphics, and Image Processing* **7**, pp. 195–210, 1978.

page 6

7. D. B. Gennery, "Least-squares camera calibration including lens distortion and automatic editing of calibration points," in *Calibration and Orientation of Cameras in Computer Vision*, A. Grün and T. S. Huang, eds., Springer-Verlag, in press.
8. J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence* **8**, pp. 679–697, Nov. 1986.
9. P. L. Rosin, "Edges: Saliency measures and automatic thresholding," *Machine Vision and Applications* **9**, pp. 139–159, 1997.
10. C. F. Olson, "Decomposition of the Hough transform: Curve detection with efficient error propagation," in *Proceedings of the European Conference on Computer Vision*, vol. 1, pp. 263–272, 1996.
11. C. F. Olson, "Efficient pose clustering using a randomized algorithm," *International Journal of Computer Vision* **23**, pp. 131–147, June 1997.
12. C. F. Olson, "An approximation algorithm for least median of squares regression," *Information Processing Letters* **63**, pp. 237–241, Sept. 1997.